WEST

Help | Logout | Interrupt

Main Menu | Search Form | Posting Counts | Show S Numbers | Edit S Numbers | Preferences | Cases

Search Results -

Terms	Documents
L19 and (relational near database or relational near data with base)	13

Database: US Patents Full-Text Database; US Pre-Grant Publication Full-Text Database; JPO Abstracts Database; EPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Search: Refine Search

Recall Text | Clear

Search History

DATE: Wednesday, March 19, 2003 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
	DB=USPT,PGPB,JPAB,EPAB,DWPI,TDBD; PLUR=YES; OP=OR		
L20	L19 and (relational near database or relational near data with base)	13	L20
L19	L18 and semantic with information	72	L19
L18	L17 and syntactic with information	124	L18
L17	hierarchy	36810	L17
L16	L15 and data same files	12	L16
L15	L14 and domain same definition	14	L15
L14	L13 and domain with object	59	L14
L13	grammar near rules	904	L13
L12	L11 and match$ same value	17	L12
L11	phonetic near value	60	L11
L10	L9 and nodes	45	L10
L9	L8 and nodes	45	L9
L8	L7 and hierarch$	73	L8
L7	L6 and character with length	170	L7
L6	L5 and value	1584	L6
L5	text near object	2325	L5
L4	l3 and text with object	2	L4
L3	syntactic near hierarch$	10	L3
L2	L1 and syntactic near hierarch$	2	L2
L1	text near object	2325	L1

END OF SEARCH HISTORY

WEST

Help | Logout | Interrupt

Main Menu | Search Form | Posting Counts | Show S Numbers | Edit S Numbers | Preferences | Cases

Search Results -

Terms	Documents
L15 and data same files	12

Database: US Patents Full-Text Database; US Pre-Grant Publication Full-Text Database; JPO Abstracts Database; EPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Search: Refine Search

Recall Text | Clear

Search History

DATE: Wednesday, March 19, 2003 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT,PGPB,JPAB,EPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L16	L15 and data same files	12	L16
L15	L14 and domain same definition	14	L15
L14	L13 and domain with object	59	L14
L13	grammar near rules	904	L13
L12	L11 and match$ same value	17	L12
L11	phonetic near value	60	L11
L10	L9 and nodes	45	L10
L9	L8 and nodes	45	L9
L8	L7 and hierarch$	73	L8
L7	L6 and character with length	170	L7
L6	L5 and value	1584	L6
L5	text near object	2325	L5
L4	l3 and text with object	2	L4
L3	syntactic near hierarch$	10	L3
L2	L1 and syntactic near hierarch$	2	L2
L1	text near object	2325	L1

END OF SEARCH HISTORY

L20: Entry 10 of 13 File: USPT Jul 15, 1997

US-PAT-NO: 5649190
DOCUMENT-IDENTIFIER: US 5649190 A

TITLE: Multi-model database system for dynamic creation and maintenance of complex objects in a real time environment

DATE-ISSUED: July 15, 1997

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Sharif-Askary; Jamshid	Palm Bay	FL		
Garje; Shashi	Melbourne	FL		
Rohela; Vijay	Melbourne	FL		
Roy; Ashok I.	Palm Bay	FL		
Yaseen; Irahim	Melbourne	FL		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Harris Corporation	Melbourne	FL			02

APPL-NO: 08/ 596691 [PALM]
DATE FILED: February 5, 1996

PARENT-CASE:
This application is a continuation of application Ser. No. 08/260,299, filed on Jun. 14,1994, now abandoned.

INT-CL: [06] G06 F 15/00, G06 F 17/28

US-CL-ISSUED: 395/612; 395/604, 395/200.02, 395/200.03, 395/200.05, 395/200.09, 364/DIG.1, 364/282.1, 364/284.4, 364/274.3
US-CL-CURRENT: 707/101; 707/4, 709/203

FIELD-OF-SEARCH: 395/200.05, 395/200.02, 395/200.03, 395/200.09, 395/604, 395/612, 264/DIG.1

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	4769772	September 1988	Dwyer	364/300
☐	5181239	January 1993	Jolissaint	379/96
☐	5191522	March 1993	Bosco et al.	364/401
☐	5212789	May 1993	Rago	395/600
☐	5226158	July 1993	Horn et al.	395/600
☐	5276885	January 1994	Milnes et al.	395/700
☐	5307484	April 1994	Baker et al.	395/600
☐	5321608	June 1994	Namba et al.	364/419.08
☐	5345586	September 1994	Hamala et al.	395/650
☐	5481700	January 1996	Thuraisingham	395/600
☐	5555408	September 1996	Fujisawa et al.	395/600
☐	5584024	December 1996	Schwartz	395/604

ART-UNIT: 237

PRIMARY-EXAMINER: Black; Thomas G.

ASSISTANT-EXAMINER: Homere; Jean R.

ABSTRACT:

A database system having a logical database, a knowledge base and a client information module is provided. The knowledge base contains knowledge regarding semantic and syntactic constraints of data relationships of data contained in the logical database. The client information module is coupled between the logical database and the knowledge base, the client information module providing an interface between a user of the database system and allowing use of the logical database according to the knowledge contained in the knowledge base. Since the knowledge necessary to maintain consistency in the database system is already contained in the knowledge base, a user can use the database system, which may contain a plurality of different relational databases, without extensive expertise in the particular databases.

12 Claims, 7 Drawing figures

(5)

L20: Entry 10 of 13 File: USPT Jul 15, 1997

DOCUMENT-IDENTIFIER: US 5649190 A
TITLE: Multi-model database system for dynamic creation and maintenance of complex objects in a real time environment

Abstract Text (1):
A database system having a logical database, a knowledge base and a client information module is provided. The knowledge base contains knowledge regarding semantic and syntactic constraints of data relationships of data contained in the logical database. The client information module is coupled between the logical database and the knowledge base, the client information module providing an interface between a user of the database system and allowing use of the logical database according to the knowledge contained in the knowledge base. Since the knowledge necessary to maintain consistency in the database system is already contained in the knowledge base, a user can use the database system, which may contain a plurality of different relational databases, without extensive expertise in the particular databases.

Brief Summary Text (10):
These and other needs are met by the present invention which provides a database system comprising a logical database; a knowledge base containing knowledge regarding semantic and syntactic constraints of data relationships of data contained in the logical database; and a client information module coupled between the logical database and the knowledge base, the client information module providing an interface between a user of the database system and allowing use of the logical database according to the knowledge contained in the knowledge base.

Detailed Description Text (2):
FIG. 1 illustrates the basic database architecture of the present invention. This architecture includes a logical database system 20, a physical database system 22 and distributed transaction processing architecture 24. The logical database system 20 provides a user with object-oriented and relational database representations of systems utilizing a frame-based knowledge representation scheme. The physical database system 22 is a real-time database having high-performance access characteristics (e.g., 1000 or more accesses per second). The real-time database 22 is a series of Unix files, for example, with specific access routines that optimize data storage and retrieval for a particular environment and applications. The real-time database 22 is generated from the logical database 20 using a data model transformation process 21 (DMTP). The logical database 20 is a source database made up of one or more commercially available relational databases. This source database represents the relational image of the real-time database 22. When data is accessed, updated, deleted, etc. the DMTP 21 converts the data from the tables typically used to represent data in commercial relational databases, to the real-time data representations of data used in performance oriented real-time databases. Such a conversion is readily programmable by one of ordinary skill in the art. This logical/physical database architecture thereby insulates the user from the underlying real-time data structures of the real-time database.

Detailed Description Text (5):
The logical database system 20 of FIG. 1 is shown in more detail in the embodiment of FIG. 2. Logical database system 20 includes a client interface module 30 that is connected to a knowledge base 32 and a logical database system engine 34. The logical database system engine 34 is connected to one or a plurality of relational databases 36, with only one such database being shown in solid lines, with possible additional databases being shown in phantom lines. The additional relational databases 36 do not all have to be the same type of relational database, and may be supplied by different vendors. These different relational databases 36 will have different toolkits and methods to generate forms, and maintain and operate the different databases. A single user would normally be faced with a daunting task to maintain and operate the different relational databases, since one ordinarily needs to understand the various interrelationships within a particular database and

corresponding toolkits. However, the logical database system 20, comprising the knowledge base 32, the logical database system engine 34 and the client interface module 30 of the present invention serve as an interface between the relational databases 36 and the user. The knowledge stored in the knowledge base 32, in conjunction with the logical database system engine 34, contains the specialized knowledge for generation, maintenance and operation of each of the different databases of the system. Thus, the user does not need to know how to use each of the individual, different relational databases.

Detailed Description Text (7):
In the knowledge base 32, a hierarchical representation of the entities and their relationships is constructed by an expert via generalizations and specializations. This hierarchical representation supports the class concept and leads to the elimination of redundant information that may be inherited from parent objects within the hierarchy. Definitions are stored with each object and describe the attributes and properties of the objects as well as procedures and methods for manipulating the object. The object hierarchy that is defined in the knowledge base 32 describes the elements of the complete system network. The knowledge base 32 therefore transforms the relational nature of the database to an object oriented nature. The system network encompasses data for the various apparatus, devices, data acquisition and control equipment, operational and procedural characteristics, and entities with temporal values of a system. (An example of a system is an electric utility power system).

Detailed Description Text (9):
A class is a collection of objects, either real or abstract, with similar characteristics. Every class in the system has one frame which is an ASCII file containing information about the class in a pre-specified format. The frame-hierarchy is organized using the object-oriented concepts of generalization-specialization and aggregation. In generalization-specialization, a class can inherit characteristics from other classes. The class that inherits characteristics is the specialization of the other and the class from which the characteristics are inherited is the generalization of the other. For example, the class Department is a specialization of the class Personnel, and the class Personnel is a generalization of the class Department.

Detailed Description Text (13):
All of the above information is contained in the knowledge base 32 which is defined as a set of frames. A frame is an ASCII file providing schema information in a pre-defined format. The basic structure of an exemplary frame is shown in FIG. 7. Each frame in the knowledge base 32 contains information about an object in the hierarchy of the database system. This information includes the class of the object, information regarding whether that object is a generalization of other objects, and whether the object is a specialization of other objects. The frame includes aggregate information, and semantic knowledge regarding the object. This semantic knowledge is that knowledge necessary to maintain the integrity of the database system as a whole, and the relational database 36 and the real-time database 32 in particular. For example, the semantic knowledge in a frame for the object or class Employee may include the rule that each employee is assigned to a Quality team and is working on a specific project. When, for example, a new attribute is added to the frame for "Employee", the semantic knowledge included in that frame is used by the database administration system to assure that this new attribute will not corrupt the relationships and constraints of the object Employee to the other objects throughout the database system.

Detailed Description Text (16):
The logical (or "source") database 36 includes a logical representation of all object instantiations. This logical database 36 contains one or more commercially available relational databases.

Detailed Description Text (20):
The view definition editor 42 is a window based point and click tool that allows an appropriately privileged user to define a "view". A view definition is a subset of the system knowledge base 32 to which some additional knowledge is added in order to handle the additional abstractness. Each parameter of this additional knowledge has a default value and the person defining the view is allowed to modify the view by choosing a value from a set of values presented to the user. Thus, a view is a customized collection of information contained in the database, as selected ("defined") by the user. Since the view definition editor uses the frames contained in the knowledge base in building a view, and since each frame contains semantic information to maintain consistency in the logical database 20, the rules necessary to maintain consistency when providing input to the source database 36 is already captured when a view is defined. The output of the view definition is a binary file.

Detailed Description Text (24):
In normal operation, a user will define a view, which includes the knowledge contained in the knowledge base 32 to maintain semantic and syntactic consistency. This view is a customized subset of the information contained in the database 36. In the company example described earlier, the view may be defined as EMPLOYEE with an instance (male) for the attribute Sex and that has a DEPENDENT. The view definition editor 42 determines whether the values provided by the user violate any rules, semantic or syntactic. Once a view has been defined, the user or other users may enter data into the database using this defined view. This data will be sent to the view engine 48 via the view data editor 44, with the appropriate semantic and syntactic checks performed, based upon the knowledge of the semantic and syntactic behavior already captured with the defined view. The view engine 48 provides the data to the source database 36 according to the appropriate rules. This data is then provided to the real-time database 22 via the data model transformation process 21, which transforms the data from the relational database structures to the structures used in the real-time database 22.

CLAIMS:

2. The database system of claim 1, wherein the logical database includes a plurality of relational databases.

5. The database system of claim 2, wherein the plurality of relational databases include different types of relational databases having different toolkits.

6. The database system of claim 2, wherein the knowledge base includes a mechanism to represent and capture structures of the relational databases using an object oriented data model.

9. The database system of claim 8, wherein each frame includes information regarding an object and semantic rules, syntactic rules, and relationships of said object to other objects in the database system.

L16: Entry 8 of 12 File: USPT Oct 31, 2000

US-PAT-NO: 6139201
DOCUMENT-IDENTIFIER: US 6139201 A

TITLE: Integrated authoring and translation system

DATE-ISSUED: October 31, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Carbonell; Jaime G.	Pittsburgh	PA		
Gallup; Sharlene L.	Morton	IL		
Harris; Timothy J.	Pekin	IL		
Higdon; James W.	Lacon	IL		
Hill; Dennis A.	East Peoria	IL		
Hudson; David C.	Edelstein	IL		
Nasjleti; David	Morton	IL		
Rennich; Mervin L.	Dunlap	IL		
Andersen; Peggy M.	Pittsburgh	PA		
Bauer; Michael M.	Pittsburgh	PA		
Busdiecker; Roy F.	Pittsburgh	PA		
Hayes; Philip J.	Pittsburgh	PA		
Huettner; Alison K.	Pittsburgh	PA		
McLaren; Bruce M.	Pittsburgh	PA		
Nirenburg; Irene	Pittsburgh	PA		
Riebling; Eric H.	Pittsburgh	PA		
Schmandt; Linda M.	Pittsburgh	PA		
Sweet; John F.	Pittsburgh	PA		
Baker; Kathryn L.	Pittsburgh	PA		
Brownlow; Nicholas D.	Pittsburgh	PA		
Holm; Susan E.	Pittsburgh	PA		
Leavitt; John Robert Russell	Pittsburgh	PA		
Lonsdale; Deryle W.	Bridgeville	PA		
Mitamura; Teruko	Pittsburgh	PA		
Nyberg, 3rd; Eric H.	Pittsburgh	PA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Caterpillar Inc.	Peoria	IL			02

APPL-NO: 08/ 632780 [PALM]
DATE FILED: April 15, 1996

PARENT-CASE:
This is a continuation application of application Ser. No. 08/363,309, filed Dec. 22, 1994.

INT-CL: [07] G06 F 3/14, G06 F 17/21

US-CL-ISSUED: 395/752; 395/758, 395/793, 395/798, 395/60
US-CL-CURRENT: 704/2; 704/8, 715/531, 715/536

FIELD-OF-SEARCH: 395/751, 395/752, 395/754, 395/757, 395/758, 395/759, 395/792, 395/793, 395/798, 395/1, 395/10, 395/12, 395/50, 395/60, 395/63

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected | Search ALL

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	4661926	April 1987	Lee	365/104
☐	4771401	September 1988	Kaufman et al.	395/795
☐	4821230	April 1989	Kumano et al.	395/756
☐	4829423	May 1989	Tennant et al.	395/759
☐	4954984	September 1990	Kaijima et al.	395/755
☐	5175684	December 1992	Chong	395/753
☐	5225981	July 1993	Yokogawa	395/752
☐	5243519	September 1993	Andrews et al.	395/758

OTHER PUBLICATIONS

Adriaens; "From Cogram to Alcogram: Toward a Controlled English Grammar Checker"; Proc. Of Coling 92; Nantes; pp. 595-601; Aug. 23, 1992.
Goodman; "The KBMT Project: A Case Study in Knowledge-Based Machine Translation"; Morgan kaufmann Publishers, Inc.; 1992.
Meyer; "Lexicographic Principals & Design for Knowledge-Based Machine Translation"; Paper No. CMU-CMT-90-118; Carnegie Mellon Center.
"KBMT-89 Project Report", Carnegie Mellon Center for Machine Translation, Feb. 1989.

Carbonell et al, "Knowledge-Based Machine Translation, the CMU Approach", Machine Translation--Theoretical and Methodological Issues.
Carbonell, Steps toward Knowledge-Based Machine Translation, IEEE Trans on Pattern Analysis and Machine Intelligence, No. 4 pp. 376-392.
Eric H. Nyberg, 3rd, "The FRAMEKIT User's Guide--Version 2.0", Carnegie Mellon Center for Machine Translation, Paper No. CMU-CMT-Memo.
Mitamura, "An Efficient Interlingua Translation System for Multi-Lingual Document Production" Wash.D.C. Jul. 2-4, 1991.
Nirenburg, Acquisition of Very Large Knowledge Bases: Methodology Tools and Applications, Carnegie Mellon, Paper No. CMU-CMT-88-108, Jun. 1988.
Nirunburg, "Machine Translation: A Knowledge Based Approach", Morgan Kaufmann Publishers, Inc. 1992.
Somers, "An Introduction to Machine Translation", Academic Press, Oct. 1991.
Teruko Mitamura, "The Hierarchical Organization of Predicate Frames for Interpretive Mapping in Natural Language Proc", #CMU-CMT-90-117.
Tomita et al, "The Universal Parser Architecture for Knowledge-Based Machine Translation", Carnegie Mellon Center for Mach. Trans.
Tomita, "Generation Kit and Transformation Kit--Version 3.2--User's Manual", Carnegie Mellon Center for Mach Translation--#CMU-CMT-88-Memo.
Tomita, "The Generalized LR Parser/Compiler Version 8.1:User's Guide" Carnegie Mellon Center for Mach Translation, Paper No. CMU-CMT-88-Memo.

ART-UNIT: 241

PRIMARY-EXAMINER: Hayes; Gail

ASSISTANT-EXAMINER: Tkacs; Stephen R.

ABSTRACT:

The present invention is a system of integrated, computer-based processes for monolingual information development and multilingual translation. An interactive text editor enforces lexical and grammatical constraints on a natural language subset used by the authors to create their text, which they help disambiguate to ensure translatability. The resulting translatable source language text undergoes machine translation into any one of a set of target languages, without the translated text requiring any postediting.

. 42 Claims, 10 Drawing figures

L16: Entry 8 of 12 File: USPT Oct 31, 2000

DOCUMENT-IDENTIFIER: US 6139201 A
TITLE: Integrated authoring and translation system

Detailed Description Text (32):
The present invention utilizes a SGML text editor 140 since it creates text using Standard Generalized Markup Language (SGML) tags. SGML is an International Standard markup language for describing the structure of electronic documents. It is designed to meet the requirements for a wide range of document processing and interchange tasks. SGML tags enable documents to be described in terms of their content (text, images, etc) and logical structure (chapters, paragraphs, figures, tables, etc.) In the case of larger, more complex, electronic documents, it also makes it possible to describe the physical organization of a document into files. SGML is designed to enable documents of any type, simple or complex, short or long, to be described in a manner that is independent of both the system and application. This independence enables document interchange between different systems for different applications without misinterpretation or loss of data.

Detailed Description Text (91):
The magnitude of the translation problems is considerably lessened by any reductions of the range of linguistic phenomena the language represents. A sublanguage covers the range of objects, processes and relations within a limited domain. Yet a sublanguage may be limited in its lexicon while it may not necessarily be limited in the power of its grammar. Under controlled situations, a strategy aimed at facilitating machine translation is that of constraining both the lexicon and the grammar of the sublanguage.

Detailed Description Text (120):
The development of a useful and complete vocabulary is important for any documentation effort. When documentation is subsequently translated, the vocabulary becomes an important resource for the translation effort. The MT 120 is designed to handle most functional items available in English, except those referring to very personal (I, me, my, etc.) or gender-based (hers, she, etc.) or other pronominal (it, them, etc.) usage. This will include a number of technical "borrowings" from English general words (such as "truck" or "length"). The vast majority of the constrained language vocabulary, then, will consist of the "special" (e.g., technical) terms of one or more words, which express the objects and processes of the special domain. To the extent that the vocabulary is able to express the full range of notions about the special domain, the vocabulary is said to be complete.

Detailed Description Text (152):
Knowledge-based Machine Translation (KBMT) must be supported by world knowledge and by linguistic semantic knowledge about meanings of lexical units and their combinations. A KBMT knowledge base must be able to represent not only a general, taxonomic domain of object types such as "car is a kind of vehicle," "a door handle is a part of a door," artifacts are characterized by (among other properties) the property "made-by"; it must also represent knowledge about particular instances of object types (e.g., "IBM" can be included into the domain model as a marked instance of the object type "corporation") as well as instances of (potentially complex) event types (e.g., the election of George Bush as president of the United States is a marked instance of the complex action "to-elect"). The ontological part of the knowledge base takes the form of a multihierarchy of concepts connected through taxonomy-building links, such as is-a, part-of, and some others. We call the resulting structure a multihierarchy because concepts are allowed to have multiple parents on each link type.

Detailed Description Text (153):
The domain model or concept lexicon contains an ontological model, which provides uniform definitions of basic categories (such as objects, event-types, relations, properties, episodes, etc.) used as building blocks for descriptions of particular domains. This "world" model is relatively static and is organized as a multiply

interconnected network ontological concepts. The gene development of an ontology of an application (sub)world in is well known in the art. See, for example, Brachman and Schmolze, An Overview of the KL-ONE Knowledge Representation System, Cognitive Science, vol. 9, 1985; Lenat, et al, Cyc: Using Common Sense Knowledge to Overcome Brittleness and Knowledge Acquisition Bottlenecks, AI Magazine, VI:65-85, 1985; Hobbs, Overview of the Tacitus Project, Computational Linguistics, 12:3, 1986; and Nirenburg et al, Acquisition of Very Large Knowledge Bases: Methodology, Tools and Applications, Center for Machine Translation, Carnegie Mellon University (1988) all of which are incorporated herein by reference.

Detailed Description Text (154):
The ontology is a language-independent conceptual representation of a specific subworld, such as heavy equipment troubleshooting and repair or the interaction between personal computers and their users. It provides the semantic information necessary in the sublanguage domain for parsing source text in interlingua text and generating target texts from interlingua texts. The domain model has to be of sufficient detail to provide sufficient semantic restrictions that eliminate ambiguities in parsing, the ontological model must provide uniform definitions of basic ontological categories that are the building blocks for descriptions of particular domains.

Detailed Description Text (161):
3. A LE Domain Model (LE/DM) 530 contains information that is required only by the LE 130; this includes non-CSL synonyms for CSL lexical items, dictionary definitions of CSL lexical items, and examples of the CSL lexical items in use.

Detailed Description Text (278):
The MT analyzer 127, guided by analysis knowledge (data files), translates a CSL text 305 input sentence in the source language into a semantic frame representation of the meaning of the sentence. The knowledge structures brought to bear in the analysis phase are the analysis grammars, the mapping rules, and the concept lexicon.

Detailed Description Text (288):
Thus the semantic component is a "visible" part of the syntactic parse. The approach, of simultaneously creating the syntactic and semantic structures, has produced a system able to eliminate "meaningless" partial parses before completing them. Semantics are added to the syntactic structure when the lexicon is accessed for the definition of a word. Another part of the definition of a word is a set of structural mapping rules. These mapping rules are used when syntactic equations in grammar rules add infirmation to a syntactic structure.

CLAIMS:

1. A computer-based system for monolingual document development, comprising:

a text editor adapted to accept interactively from an author input text written in a source language;

a language editor, which is an extension of said text editor, which interactively enforces lexical constraints and grammatical constraints on a natural language subset used by said author to create said input text, wherein said author is interactively aided in enforcing said lexical constraints and said grammmatical constraints on said input text so as to produce unambiguous constrained text;

a machine translation system, responsive to said language editor that is configured to translate said unambiguous constrained text into a foreign language; and

a domain model, which communicates with said language editor, wherein said domain model provides pre-determined domain knowledge and linguistic semantic knowledge about lexical units and of their combinations, so as to assist said language editor in said enforcement of said lexical and grammatical constraints wherein said domain model is a tripartite domain model, said tripartite domain model comprising,

a kernel which contains lexical information that is required by said language editor and said machine translation system, wherein said lexical information includes lexical items within said natural language subset along with associated semantic concepts, parts of speech, and morphological information,

a language editor domain model which contains information that is required only by said language editor, wherein said information includes at least one of a natural language subset of synonyms for items not within said natural language subset, a dictionary of definitions of said lexical items, and a set of examples of using said

lexical items, and

a machine translation domain model which contains information which is required by only said machine translation system, said machine translation domain model includes a hierarchy of concepts used for unambiguous mapping and semantic verification in translation.

2. A computer-based system for monolingual document development, comprising:

a text editor adapted to accept interactively from a author information elements written in a source language;

a language editor, which is an extension of said text editor, which interactively enforces lexical and grammatical constraints on a natural language subset used by said author to create unambiguous constrained information elements, wherein said author interactively aids in enforcing said lexical and grammatical constraints on said input text so as to produce said unambiguous constrained information elements;

memory means for storing said unambiguous constrained information elements for subsequent use;

a machine translation system, responsive to said language editor, that is configured to translate said unambiguous constrained information elements into a foreign language; and

a domain model, which communicates with said language editor, wherein said domain model provides pre-determined domain knowledge and linguistic semantic knowledge about lexical units and of their combinations, so as to assist said language editor in said enforcement of said lexical and grammatical constraints wherein said domain model is a tripartite domain model, said tripartite domain model comprising,

a kernel which contains lexical information that is required by said language editor and said machine translation system, wherein said lexical information includes lexical items within said natural language subset along with associated semantic concepts, parts of speech, and morphological information,

a language editor domain model which contains information that is required only by said language editor, wherein said information includes at least one of a natural language subset of synonyms for items not within said natural language subset, a dictionary of definitions of said lexical items, and a set of examples of using said lexical items, and

a machine translation domain model which contains information which is required by only said machine translation system, said machine translation domain model includes a hierarchy of concepts used for unambiguous mapping and semantic verification in translation.

3. A computer-based method for monolingual document development, comprising the steps of:

(1) entering input text in a source language into a text editor

(2)checking, via a language editor, said input text against a pre-determined set of constraints stored in a domain model that provides pre-determined domain knowledge and linguistic semantic knowledge about lexical units and of their combinations, said pre-determined set of constraints includes a set of source sublanguage rules concerning vocabulary and grammar, wherein said domain model is a tripartite domain model, said tripartite domain model comprising,

a kernel which contains lexical information that is required by said language editor and a machine translation system, wherein said lexical information includes lexical items that satisfy said pre-determined set of constraints along with associated semantic concepts, parts of speech, and morphological information,

a language editor domain model which contains information that is required only by said language editor, wherein said information includes at least one of a subset of synonyms for items that do not satisfy said pre-determined set of constraints, a dictionary definitions of said lexical items, and a set of examples of using said lexical items, and

a machine translation domain model which contains information which is required by only said machine translation system, said machine translation domain model includes a hierarchy of concepts used for unambiguous mapping and semantic verification in

translation;

(3) providing to an author interactive feedback relating to said input text, said interactive feedback indicating if said pre-determined set of constraints is met, said interactive feedback is performed subsequent to referring to said domain model which provides the necessary domain knowledge and linguistic semantic knowledge about lexical units and of their combinations, and grammar of a subset of a natural language; and

(4) producing, after completion of step (3), unambiguous constrained text.

5. A computer-based method for monolingual document development, comprising the steps of:

(1) entering input text in a source language into a text editor;

(2) checking, via a language editor, said input text against a constrained source language;

(3) providing to an author interactive feedback relating to said source input text if non-constrained source language is present in said source input text until said author modifies said source input text into a constrained source text, said interactive feedback is performed after consulting a domain model which provides the necessary domain knowledge and linguistic semantic knowledge about lexical units and of their combinations, wherein said domain model is a tripartite domain model, said tripartite domain model comprising,

a kernel which contains lexical information that is required by said language editor and said a machine translation system, wherein said lexical information includes lexical items within said constrained source language along with associated semantic concepts, parts of speech, and morphological information,

a language editor domain model which contains information that is required only by said language editor, wherein said information includes at least one of a natural language subset of synonyms for items not within said constrained source language, a dictionary definitions of said lexical items, and a set of examples of using said lexical items, and

a machine translation domain model which contains information which is required by only said machine translation system, said machine translation domain model includes a hierarchy of concepts used for unambiguous mapping and semantic verification in translation;

(4) checking for syntactic grammatical errors and semantic ambiguities in said constrained source text by consulting said domain model; and

(5) providing to said author interactive feedback to remove said syntactic grammatical errors and said semantic ambiguities in said constrained source text to produce unambiguous constrained text.

6. A computer-based method for monolingual document development, comprising the steps of:

(1) entering into a text editor at least one information element created in a source language;

(2) checking, via a language editor, said at least one information element against a constrained source language;

(3) providing to an author interactive feedback relating to said at least one information element if non-constrained source language is present in said at least one information element until said at least one information element has been modified into a constrained source text, said interactive feedback is performed after referring to a domain model which provides the necessary domain knowledge and linguistic semantic knowledge about lexical units and their combinations, wherein said domain model is a tripartite domain model, said tripartite domain model comprising:

a kernel which contains lexical information that is required by said language editor and said a machine translation system, wherein said lexical information includes lexical items within said constrained source language along with associated semantic concepts, parts of speech, and morphological information,

a language editor domain model which contains information that is required only by said language editor, wherein said information includes at least one of a natural language subset synonyms for items not within said constrained source language, a dictionary of definitions of said lexical items, and a set of examples of using said lexical items, and

a machine translation domain model which contains information which is required by only said machine translation system, said machine translation domain model includes a hierarchy of concepts used for unambiguous mapping and semantic verification in translation;

(4) checking for syntactic grammatical errors and semantic ambiguities in said constrained source text by consulting said domain model;

(5) providing interactive feedback to said author to remove said syntactic grammatical errors and said semantic ambiguities in said constrained source text to produce at least one unambiguous constrained information element; and

(6) saving said at least one unambiguous constrained information element for later use.

7. A computer-based system for translating source language input text to a foreign language, comprising:

a text editor adapted to accept interactively from an author the input text written in a source language;

a language editor, which is an extension of said text editor, which interacts with said author to produce from said input text an unambiguous constrained source text by interactively enforcing vocabulary and grammatical constraints against a constrained source language;

a machine translation system, responsive to said language editor, which is configured to translate said unambiguous constrained source text into the foreign language; and

a domain model, which communicates with said language editor and said machine translation system, and which provides predetermined domain knowledge and linguistic semantic knowledge about lexical units and of their combinations, so as to aid in producing said unambiguous constrained source text and in said translation to the foreign language, wherein said domain model is a tripartite domain model, said tripartite domain model comprising,

a kernel which contains lexical informationthat is required by said language editor and said machine translation system, wherein said lexical information includes lexical items within said constrained source language along with associated semantic concepts, parts of speech, and morphological information,

a language editor domain model which contains information that is required only by said language editor, wherein said information includes at least one of a subset of synonyms for items not within said constrained source language, a dictionary definitions of said lexical items, and a set of examples of using said lexical items, and

a machine translation domain model which contains information which is required by only said machine translation system, said machine translation domain model includes a hierarchy of concepts used for unambiguous mapping and semantic verification in translation.

24. A computer-based system for monolingual document development and multilingual translation, comprising:

a text editor adapted for accepting interactively from an author information elements written in a source language;

a language editor, which is an extension of said text editor, which interactively enforces lexical and grammatical constraints on a natural language subset used by said author to create said input text, wherein said author is interactively aided in enforcing said lexical and grammatical constraints on said information elements to produce said unambiguous constrained information elements;

machine translation system, responsive to said language editor, which translates said unambiguous constrained information elements into a foreign language; and

a domain model, which communicates with said language editor and said machine translation means, wherein said domain model provides pre-determined domain knowledge and linguistic semantic knowledge about lexical units and their combinations, so as to aid in producing said unambiguous constrained source text and in said translation to said foreign language, wherein said domain model is a tripartite domain model, said tripartite domain model comprising,

a kernel which contains lexical information that is required by said language editor and said a machine translation system, wherein said lexical information includes lexical items within said natural language subset along with associated semantic concepts, parts of speech, and morphological information,

a language editor domain model which contains information that is required only by said language editor, wherein said information includes at least one of a natural language subset of synonyms for items not within said natural language subset, a dictionary definitions of said lexical items, and a set of examples of using said lexical items, and

a machine translation domain model which contains information which is required by only said machine translation system, said machine translation domain model includes a hierarchy of concepts used for unambiguous mapping and semantic verification in translation.

25. A computer-based system for monolingual document development and multilingual translation, comprising:

(A) a text editor adapted to accept interactively from an author input text written in a source language;

(B) a language editor, which is an extension of said text editor, which interactively enforces lexical and grammatical constraints on a natural language subset used by said author to create said input text, said language editor comprising,

(i) a vocabulary checker which identifies occurrences of words that do not conform to said lexical constraints and which interactively aids said author in finding valid lexical replacements for said words that do not conform, and

(ii) a grammar checker which provides interactive feedback to said author concerning syntactic and semantic ambiguity, said interactive feedback producing unambiguous constrained text; and

(C) a domain model which communicates with said language editor, wherein said domain model provides pre-determined domain knowledge and linguistic semantic knowledge about lexical units and their combinations; and

(D) a machine translation system, responsive to said language editor, which is configured to translate said unambiguous constrained text into a foreign language,;

wherein said domain model is a tripartite domain model, said tripartite domain model comprising,

a kernel which contains lexical information that is required by said language editor and said a machine translation system, wherein said lexical information includes lexical items within said natural language subset along with associated semantic concepts, parts of speech, and morphological information,

a language editor domain model which contains information that is required only by said language editor, wherein said information includes at least one of a natural language subset of synonyms for items not within said natural language subset, a dictionary definitions of said lexical items, and a set of examples of using said lexical items, and

a machine translation domain model which contains information which is required by only said machine translation system, said machine translation domain model includes a hierarchy of concepts used for unambiguous mapping and semantic verification in translation.

26. A computer-based method for translating source language text to a foreign language, comprising the steps of:

(1) entering input text in a source language into a text editor;

(2) checking, via a language editor, said input text against a constrained source language;

(3) providing to an author interactive feedback relating to said source input text if nonconstrained source language is present in said source input text until said author modifies said source input text into a constrained source text, said interactive feedback includes allowing said author to select, from a list of at least one synonym, a word or phrase to replace said nonconstrained source language;

(4) checking for syntactic grammatical errors and semantic ambiguities in said constrained source text;

(5) providing interactive feedback to said author to remove said syntactic grammatical errors and said semantic ambiguities in said constrained source text to produce unambiguous constrained source text; and

(6) translating, via a machine translation system, said unambiguous constrained source text into a target language;

wherein steps (2) and (4) further include the step of communicating with a tripartite domain model (DM), wherein said tripartite DM provides predetermined domain knowledge and linguistic semantic knowledge about lexical units and their combinations, said tripartite domain model including,

a kernel which contains lexical information that is required by said language editor and said a machine translation system, wherein said lexical information includes lexical items within said constrained source language along with associated semantic concepts, parts of speech, and morphological information,

a language editor domain model which contains information that is required only by said language editor, wherein said information includes at least one of a set of synonyms for items not within said constrained source language, a dictionary of definitions of said lexical items, and a set of examples of using said lexical items, and

a machine translation domain model which contains information which is required by only said machine translation system, said machine translation domain model includes a hierarchy of concepts used for unambiguous mapping and semantic verification in translation.

37. A computer-based method for monolingual document development and multilingual translation, comprising the steps of:

(1) entering input text in a source language into a text editor;

(2) checking, via a language editor, said input text against a predetermined set of constraints stored in a domain model, wherein said predetermined set of constraints includes a set of source sublanguage rules concerning vocabulary and grammar, wherein said interactive feedback is performed in order to make said input text conform with said set of source sublanguage rules and to eliminate ambiguities, wherein said domain model is a tripartite domain model, said tripartite domain model comprising,

a kernel which contains lexical information that is required by said language editor and said a machine translation system, wherein said lexical information includes lexical items that satisfy said pre-determined set of constraints along with associated semantic concepts, parts of speech, and morphological information,

a language editor domain model which contains information that is required only by said language editor, wherein said information includes at least one of a set of synonyms for items that do not satisfy said pre-determined set of constraints, a dictionary of definitions of said lexical items, and a set of examples of using said lexical items, and

a machine translation domain model which contains information which is required by only said machine translation system, said machine translation domain model includes a hierarchy of concepts used for unambiguous mapping and semantic verification in translation;

(3) providing to an author interactive feedback relating to said input text if said predetermined set of criteria is not met, said interactive feedback is performed subsequent to consulting said domain model which provides the necessary domain

knowledge and linguistic semantic knowledge about lexical units and their combinations, wherein said author produces, through said interactive feedback, unambiguous constrained source text;

(4) translating said unambiguous constrained source text into a target language.

39. A computer-based method for monolingual document development and multilingual translation, the computer-based method comprising the steps of:

(1) entering input text in a source language into a text editor;

(2) checking, via a language editor, said input text against a constrained source language;

(3) providing to an author interactive feedback relating to said source input text if nonconstrained source language is present in said source input text until said source input text has been modified into a constrained source text, said interactive feedback being done subsequent to consulting a domain model which provides the necessary domain knowledge and linguistic semantic knowledge about lexical units and their combinations, wherein said domain model is a tripartite domain model, said tripartite domain model comprising,

a kernel which contains lexical information that is required by said language editor and said a machine translation system, wherein said lexical information includes lexical items within said constrained source language along with associated semantic concepts, parts of speech, and morphological information,

a language editor domain model which contains information that is required only by said language editor, wherein said information includes at least one of a natural language subset of synonyms for items not within said constrained source language, a dictionary of definitions of said lexical items, and a set of examples of using said lexical items, and

a machine translation domain model which contains information which is required by only said machine translation system, said machine translation domain model includes a hierarchy of concepts used for unambiguous mapping and semantic verification in translation;

(4) checking for syntactic grammatical errors and semantic ambiguities in said constrained source text by consulting said domain model;

(5) providing interactive feedback to said author to remove said syntactic grammatical errors and said semantic ambiguities in said constrained source text to produce at lease one unambiguous constrained source text; and

(6) saving said at least one unambiguous constrained information element for later use;

(7) translating with said machine translation system said at least one unambiguous constrained source text into a foreign language.

40. A computer-based method for monolingual document development and multilingual translation, comprising the steps of:

(1) entering into a text editor at least one information element created in a source language;

(2) checking, via a language editor, said at least one information element against a constrained source language;

(3) providing to an author interactive feedback relating to said at least one information element if nonconstrained source language is present in said at least one information element until said at least one information element has been modified into a constrained source test, said interactive feedback is performed after consulting a domain model which provides the necessary domain knowledge and linguistic semantic knowledge about lexical units and of their combinations, wherein said domain model is a tripartite domain model, said tripartite domain model comprising,

a kernel which contains lexical information that is required by said language editor and said a machine translation system, wherein said lexical information includes lexical items within said natural language subset along with associated semantic concepts, parts of speech, and morphological information,

a language editor domain model which contains information that is required only by said language editor, wherein said information includes at least one of a natural language subset of synonyms for items not within said natural language subset, a dictionary of definitions of said lexical items, and a set of examples of using said lexical items, and

a machine translation domain model which contains information which is required by only said machine translation system, said machine translation domain model includes a hierarchy of concepts used for unambiguous mapping and semantic verification in translation;

(4) checking for syntactic grammatical errors and semantic ambiguities in said constrained text by consulting said domain model;

(5) providing interactive feedback to said author to remove said syntactic grammatical errors and said semantic ambiguities in said constrained source text to produce at least one unambiguous constrained information element;

(6) saving said at least one unambiguous constrained information element for later use;

(7) translating with said machine translation system said at least one unambiguous constrained information element into a foreign language.

(2)

WEST

Generate Collection | Print

L12: Entry 12 of 17 File: USPT Feb 7, 1995

US-PAT-NO: 5388257
DOCUMENT-IDENTIFIER: US 5388257 A

TITLE: Method and apparatus for operating a computer based file system

DATE-ISSUED: February 7, 1995

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Bauer; Eric J.	Tinton Falls	NJ		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
AT&T Corp.	Murray Hill	NJ			02

APPL-NO: 08/ 247619 [PALM]
DATE FILED: May 23, 1994

PARENT-CASE:
This application is a continuation of application Ser. No. 08/74690, filed on Jun. 10, 1993, which is a continuation of application Ser. No. 07/735393, filed on Jul. 24, 1991, both abandoned.

INT-CL: [06] G06 F 15/40

US-CL-ISSUED: 395/600; 395/400, 395/425, 395/700, 364/DIG.1, 364/282.3, 364/282.4, 364/280
US-CL-CURRENT: 707/1

FIELD-OF-SEARCH: 395/400, 395/425, 395/600, 395/700

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected | Search ALL

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	4315310	February 1982	Bayliss et al.	364/200
☐	4325120	April 1982	Colley et al.	364/200
☐	4654779	March 1987	Kato et al.	395/400
☐	4768150	August 1988	Chang et al.	395/700
☐	4780821	October 1988	Crossley	395/650
☐	4823306	April 1989	Barbic et al.	364/900
☐	4941084	July 1990	Terada et al.	395/650
☐	5006992	April 1991	Skeirik	364/513
☐	5051947	September 1991	Messenger et al.	364/900
☐	5109515	April 1992	Laggis et al.	395/725
☐	5182766	January 1993	Garland	379/216
☐	5202828	April 1993	Vertelney et al.	364/419
☐	5206951	April 1993	Khoyi et al.	395/650
☐	5218696	June 1993	Baird et al.	395/600
☐	5313646	May 1994	Hendricks et al.	395/600

OTHER PUBLICATIONS

Peterson, "A Yellow-Pages Service for a Local-Area Network", Computer Communication Review, pp. 235-242 ACM 0-89791-245-4/88/0001/0235, 1988.
Tannenbaum, Andrew J., Computer Networks, Prentice-Hall, Inc. 1981.

ART-UNIT: 237

PRIMARY-EXAMINER: Black; Thomas G.

ASSISTANT-EXAMINER: Amsbury; Wayne

ABSTRACT:

A computer-based file system accesses files with search requests evoked by file identifiers that are embedded in standard operating system calls. The file identifiers can be different from the standard operating system hierarchical pathname of the file, and it can include criterion type/value pairs, multiple syntax structures, substrings of the files, database calls, and phonetic spellings.

33 Claims, 8 Drawing figures

L12: Entry 12 of 17 File: USPT Feb 7, 1995

DOCUMENT-IDENTIFIER: US 5388257 A
TITLE: Method and apparatus for operating a computer based file system

Brief Summary Text (9):
According to my invention, a computer-based file apparatus permits a real-time user-selectable search request of previously stored data objects using file access system calls which use search criteria other than a file-name-substring matching. Search criteria includes a criterion type and a criterion value which combination is also referred to as a search criterion type/value pair. The file access system call includes a purported file name containing one or more non-file-name-substring-based search criteria. Using my non-file-name-substring-based search technique, a user can do look-up-by-key, relational look-up, phonetic spelling look-up, and find-operator look-up to locate one or more stored data objects of any database or file system of the apparatus. A user may be a person, a program or an apparatus desiring to access stored data objects. According to one feature, a key associated with each located data object is returned to the user. The search criteria may identify single-data object or multiple-data object. Another feature enables multiple-data objects to be organized in a virtual directory to facilitate subsequent data object access by the user.

Detailed Description Text (21):
According to another feature of the present invention, there may be many types of exotic look-up criteria. Each search criteria includes a criteria type (e.g., "phonetic" of 333 in FIG. 3) and a criteria value (e.g., "Kacurro" of 333 in FIG. 3) also referred to herein as a search criteria type/value pair. As previously discussed, existing file systems support file name matching either by exact character matching of file names, or by case-insensitive matching, or by some form of substring matching (e.g., "fo?", "f*"). Henceforth, these exotic types of file name matching are referred to as "non-conventional" file name matching. Thus, in addition to conventional matching, this invention covers non-conventional look-up criteria (i.e., non-file-name-substring-based search criteria) for which the purported file name is neither an exact character, case-insensitive, nor substring of the target file name. Non-conventional look-up criteria fall into two broad categories: 1) single-object criteria that can match 0 or 1 objects; 2) multi-object criteria that can match 0, 1, or more objects. A single-object criterion may be a characteristic of the desired data object, such as a unique identifier or key to the desired data object (e.g., social security number). Conventional file name matching is a single-object criterion because the file system guarantees that a particular file name is unique within a directory. A look-up by inode number (also referred to as ino) is a single-object criterion because inodes are defined to be unique. A look-up to identify all objects on a file system owned by a particular user or to identify all files held in a particular directory is a multi-object criterion because several objects could simultaneously meet these criteria.

Detailed Description Text (25):
In the present invention, the results of a multi-object criterion look-up are returned via the get directory entry (getdents()) system call. Each object is returned with a file name and a unique access key as the inode of the entry. Given the unique access key (e.g., inumber) of each target entry, a user can then use a single-object criterion, such as "ino=decimal.sub.-- inumber," to access the target entry. In this case, the back-door syntax uses "ino=" as a prepended segment to signal that the inode number lookup search criteria should be used, and the "decimal.sub.-- inumber" is a base name which gives the decimal representation of the inode key that selects the correct target entry. These directories are "virtual" in the sense that they are created on demand when a user asks to see the contents of the directory, and they are automatically removed when the user is done with it. For example, consider using this invention to implement an electronic phone book. An application issues a phonetic look-up by interrogating the directory "/phonebook/phonetic=kacurro." In this case, the back-door syntax uses "phonetic=" as a prepended segment to signal that the phonetic matching search criteria should

be used, and "kaccuro" a base name which gives the phonetic value to match against. Successive calls to the get directory entry system call might return: "caccuro,annemarie" with inumber 745; and "caccuro,john" with inumber 1124. The user would select "caccuro,john" and the application would access that entry by opening "/phonebook/ino=1124." Note that because virtual directories are being created and manipulated, an implementation could return several entries with identical file names but different inumbers, such as "smith,john."

Detailed Description Text (47):
In step 601 the system parses the inode number (ino) value X from the file name (ino=X, gen=Y). In step 602 the system parses the generation (gen) number Y from the file name. In step 604 the system accesses the inode list, 223, to fetch the file identified by inode number (ino) X. Note inode number value X refers to a vnode associated therewith. In step 606 the system determines a valid data object was fetched. If not, then in step 608 a no match or false condition is returned to step 527 of FIG. 5.

Detailed Description Text (52):
In step 702 the file name is parsed to extract the requested organization name "orgname". In step 704 the data base or stored items are accessed to determine if "orgname" is a valid data object grouping. If not, then "no match" return is returned in step 706 (to step 535 of FIG. 5). If an "orgname" data object grouping is valid, then in step 708 a vnode is allocated for a virtual directory which will store all the data objects under the grouping "orgname". In step 710 the vnode is initialized in the normal manner. In step 712 all of the data object (or files) matching the "orgname" criterion are located. In step 714 a data buffer is allocated. In step 716 the name and key values (inode numbers, ino) are written into data buffer in directory format (as in FIG. 8). In step 718 the data buffer is linked to the virtual directory vnode. In step 720 an indication that the match was successful is returned to step 513 of FIG. 5.

CLAIMS:

8. The apparatus of claim 1 wherein at least one search criterion value is a data substring which is to be matched against the contents of said data files.

10. The apparatus of claim 1 wherein the file identifier includes a base name segment and one or more appended segments and wherein said locating means utilizes at least one of said appended segments as a search criterion type and uses said base name segment as a matching search criterion value to locate said one or more data files.

13. A client/server network including a server apparatus arranged to communicate over a facility to one or more client apparatuses, said server apparatus storing a plurality of previously-stored data files, each identified by a filename, wherein

each client apparatus includes means for generating and transmitting a file identifier comprised of a string of characters as part of a standard operating system call to said server apparatus, said file identifier being different from the standard operating system hierarchical pathname for said desired data file, said file identifier containing one or more search criterion type/value pairs, each search criterion type/value pair having a search criterion type and value, each search criterion type/value pair being matched directly against a characteristic of the previously-stored data files, and

said server apparatus includes a computer-based file apparatus for directly accessing any of the previously-stored data files using at least one of said search criterion type/value pairs, each search criterion type/value pair being matched directly against a characteristic of the previously stored data files, the accessing being performed in the absence of requiring presorting of said data files.

27. The method of claim 20 wherein at least one search criterion value is a data substring which is to be matched against the contents of said data files.

WEST

(2)

Generate Collection | Print

L10: Entry 10 of 45 File: USPT Sep 24, 2002

US-PAT-NO: 6457013
DOCUMENT-IDENTIFIER: US 6457013 B1

TITLE: Data formating property modifiers

DATE-ISSUED: September 24, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Saxton; Catherine J.	Issaquah	WA		
Saxton; Thomas W.	Issaquah	WA		
Walker; Mark John	Redmond	WA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Microsoft Corporation	Redmond	WA			02

APPL-NO: 09/ 172414 [PALM]
DATE FILED: October 14, 1998

PARENT-CASE:
RELATED APPLICATIONS The present application is a divisional of non-provisional patent application entitled "Forward Extensible Property Modifiers for Formatting Information in a Program Module", filed on Jul. 16, 1997 and assigned U.S. application Ser. No. 08/893,939 now U.S. Pat. No. 6,016,492.

INT-CL: [07] G06 F 6/00

US-CL-ISSUED: 707/101; 707/104
US-CL-CURRENT: 707/101

FIELD-OF-SEARCH: 707/1-225

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected | Search ALL

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	5446575	August 1995	Lysakowski	
☐	5542078	July 1996	Martel et al.	
☐	5627972	May 1997	Shear	395/200.76
☐	5692157	November 1997	Williams	395/500
☐	5794042	August 1998	Terada et al.	395/701

ART-UNIT: 2171

PRIMARY-EXAMINER: Black; Thomas G.

ASSISTANT-EXAMINER: Mills; John G.

ABSTRACT:

Property modifiers stored on a computer-readable medium for formatting information associated with a computer file, such as an electronic document. Each property modifier includes a size field having information indicating the size of its property modifier argument and an index field having an index into a property information array. A program module can traverse a group of property modifiers by utilizing the information in the size field of the current property modifier to determine the location of the next property modifier. Formatting is applied to information of the computer file by using the index of each property modifier to access formatting information maintained in the property information array. Each property modifier can store encoded information corresponding to the type of object to receive formatting, the size of the property modifier's argument, and an index into the property information array associated with an executable computer program. Rather than pre-allocate space in the executable portion of a program module, the property modifier itself contains information embedded therein that indicates the size of the parameter of the property modifier.

17 Claims, 8 Drawing figures

L10: Entry 13 of 45 File: USPT Apr 16, 2002

US-PAT-NO: 6374252
DOCUMENT-IDENTIFIER: US 6374252 B1

TITLE: Modeling of object-oriented database structures, translation to relational database structures, and dynamic searches thereon

DATE-ISSUED: April 16, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Althoff; James	Los Altos	CA		
Lee; Seung	San Jose	CA		
Belanger; Ken	Hudson	NH		
Prasad; Nagendra	Padmanabhangar			IN
McGinnis; Brian	Danville	CA		
McWilliams; Floyd	Palo Alto	CA		
Zhang; Yong	Sunnyvale	CA		
Koushik; Ravi	Mountain View	CA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
i2 Technologies US, Inc.	Dallas	TX			02

APPL-NO: 08/ 951714 [PALM]
DATE FILED: October 16, 1997

PARENT-CASE:
CROSS-REFERENCE TO RELATED APPLICATION This is a file-wrapper continuation of U.S. patent application Ser. No. 08,521,667, filed Aug. 31, 1995 now abandoned which is a continuation of 08/428,003 filed Apr. 24, 1995 abandoned.

INT-CL: [07] G06 F 17/30

US-CL-ISSUED: 707/102; 707/100, 707/103
US-CL-CURRENT: 707/102; 707/100, 707/103R

FIELD-OF-SEARCH: 707/1, 707/103, 707/102, 707/100, 707/101, 707/104

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	4930071	May 1990	Tou et al.	364/300
☐	5133075	July 1992	Risch	395/800
☐	5161225	November 1992	Abraham et al.	395/600
☐	5181162	January 1993	Smith et al.	364/419
☐	5201046	April 1993	Goldberg et al.	395/600
☐	5212787	May 1993	Baker et al.	395/600
☐	5235701	August 1993	Ohler et al.	395/600
☐	5261080	November 1993	Khoyi et al.	395/500
☐	5291583	March 1994	Bapat	395/500
☐	5295256	March 1994	Bapat	395/600
☐	5295261	March 1994	Simonetti	395/600
☐	5303379	April 1994	Khoyi et al.	395/700
☐	5317742	May 1994	Bapat	395/700
☐	5398336	March 1995	Tantry et al.	395/600
☐	5421015	May 1995	Khoyi et al.	395/650
☐	5426780	June 1995	Gerull et al.	707/103
☐	5437027	July 1995	Bannon et al.	707/103
☐	5448727	September 1995	Annevelink	707/7
☐	5499371	March 1996	Henninger et al.	395/700
☐	5504885	April 1996	Alashqur	707/103
☐	5548749	August 1996	Kroenke et al.	395/600
☐	5596746	January 1997	Shen et al.	395/612
☐	5615362	March 1997	Jensen et al.	395/614
☐	5765159	June 1998	Srinivasan	707/102
☐	5799309	August 1998	Srinivasan	707/102
☐	5873093	February 1999	Williamson et al.	707/103
☐	6122641	September 2000	Williamson et al.	707/103
☐	6223227	April 2001	Willianson et al.	709/315

OTHER PUBLICATIONS

R.G.G. Cattell, "Object Data Management--Object-Oriented and Extended Relational Database Systems", bibliography, pp. 273-310.
"Object-Oriented Database Management--Applications in Engineering and Computer Science", bibliography, pp. 631-662.
Donald K. Burleson, "Practical Application of Object-Oriented Techniques to Relational Databases", bibliography, pp. 243-247.
Setrag Khoshafian, "Object-Oriented Databases", bibliography, pp. 335-350.
Kroenke, Business Modeling with Semantic Objects, Wall Data Inc, pp. 1-4, Dec. 1994.

ART-UNIT: 2171

PRIMARY-EXAMINER: Coby; Frantz

ABSTRACT:

A method and system for modeling of object-oriented database structures, translation to relational database structures, and dynamic searches thereon. The user may create, edit and manipulate a user's object database (dynamically translated into a set of relational database structures), to create, edit and manipulate objects for that object database (dynamically translated into data for those relational database structures), and to create, edit and manipulate queries to be applied to that object

database (dynamically translated into queries to be applied to those relational database structures). A meta-model of the user's object database, which is itself an object database, and which has itself been translated into a set of relational database structures for manipulation by a relational database engine. The meta-model comprises a set of classes, objects, and relationships between classes which model the classes and relationships between classes of the system. Each of these classes may comprise a set of searchable properties, and each of these relationships may comprise an inheritance relationship (between a base class and a derived class) or a data-model relationship (such as a one-to-one, one-to-many, or many-to-many relationship). The data model of the user's object database is modeled by actual objects in the meta-model, and editing or manipulating the user's object database is modeled by creating, modifying, or deleting objects in the meta-model. The meta-model also models itself, in the same manner as it models the user's object database, and may be manipulated in the same manner as the user's object database.

36 Claims, 10 Drawing figures

WEST

Generate Collection Print

L3: Entry 3 of 10 File: USPT Oct 23, 2001

US-PAT-NO: 6308172
DOCUMENT-IDENTIFIER: US 6308172 B1

TITLE: Method and apparatus for partitioning a database upon a timestamp, support values for phrases and generating a history of frequently occurring phrases

DATE-ISSUED: October 23, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Agrawal; Rakesh	San Jose	CA		
Srikant; Ramakrishnan	San Jose	CA		
Lent; Brian Scott	Union City	CA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
International Business Machines Corporation	Armonk	NY			02

APPL-NO: 09/ 348595 [PALM]
DATE FILED: July 6, 1999

PARENT-CASE:
This is a continuation of U.S. patent application Ser. No. 08/909,901, filed Aug. 12, 1997 which issued as U.S. Pat. No. 6,006,223 on Dec. 21, 1999.

INT-CL: [07] G06 F 17/30

US-CL-ISSUED: 707/5; 707/2, 707/6, 707/100, 707/102, 707/203, 707/511, 707/536, 704/4, 704/8, 704/9
US-CL-CURRENT: 707/5; 704/4, 704/8, 704/9, 707/100, 707/102, 707/2, 707/203, 707/6, 715/511, 715/536

FIELD-OF-SEARCH: 707/1-10, 707/100-104, 707/200-206, 707/500, 707/511, 707/536, 704/1-10, 704/205, 704/221-223, 704/240-245, 704/251-252, 704/257, 704/267-268, 704/276, 706/45-52, 711/118-123, 712/2, 712/12-13, 712/240, 714/20

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	5768603	June 1998	Brown et al.	704/9
☐	5790848	August 1998	Wlaschin	707/3
☐	5794178	August 1998	Caid et al.	704/9
☐	6006223	December 1999	Agrawal et al.	707/5

ART-UNIT: 217

PRIMARY-EXAMINER: Choules; Jack

ASSISTANT-EXAMINER: Channavajjala; Srirama

ABSTRACT:

A method and apparatus for mining text databases, employing sequential pattern phrase identification and shape queries, to discover trends. The method passes over a desired database using a dynamically generated shape query. Documents within the database are selected based on specific classifications and user defined partitions. Once a partition is specified, transaction IDs are assigned to the words in the text documents depending on their placement within each document. The transaction IDs encode both the position of each word within the document as well as representing sentence, paragraph, and section breaks, and are represented in one embodiment as long integers with the sentence boundaries. A maximum and minimum gap between words in the phrases and the minimum support all phrases must meet for the selected time period may be specified. A generalized sequential pattern method is used to generate those phrases in each partition that meet the minimum support threshold. The shape query engine takes the set of phrases for the partition of interest and selects those that match a given shape query. A query may take the form of requesting a trend such as "recent upwards trend", "recent spikes in usage", "downward trends", and "resurgence of usage". Once the phrases matching the shape query are found, they are presented to the user.

9 Claims, 9 Drawing figures

L3: Entry 3 of 10 File: USPT Oct 23, 2001

DOCUMENT-IDENTIFIER: US 6308172 B1
TITLE: Method and apparatus for partitioning a database upon a timestamp, support values for phrases and generating a history of frequently occurring phrases

Detailed Description Text (18):
In step 312, "pruning" of the phrases which meet the requirements of the shape query may be performed. Pruning refers to the elimination of phrases which are not of interest to the user. and are deemed "uninteresting". If prunina is desired. in step 314 the pruning may comprise dropping non-maximal phrases when their support is near that of a maximal phrase that is a superset of the phrases discovered. A maximal phrase is a phrase that has maximum support in the data partition. In another embodiment, the pruning of step 314 may involve the use of a syntactic hierarchial ordering of phrases. The idea is that if a phrase X is a syntactic subphrase of a phrase Y, then the concept corresponding to X is usually a generalization of the concept corresponding to phrase Y. Such an ordering allows users to explore lower-level concepts by selecting some of the non-maximal phrases, being that users of the invention would initially see only the most general concepts. Regardless of whether pruning in step 314 occurs or not, the results of the database mining of the method 300 are displayed in step 316. The results may be displayed on various mediums as described above relative to output module 118 of FIG. 1. The method ends in step 318.

Detailed Description Text (27):
The top phrases found for U.S. Patents in this category, classification 376, were generated using the pruning techniques discussed earlier in this application. As can be seen from FIG. 7A and FIG. 7B, the support value for each phrase is shown as a percentage in the left hand column with the 0-phrase represented in the right hand column. FIG. 7B shows the results of a user-specified ordering on the phrases in FIG. 7A. The ordering of FIG. 7B included a pruning step where the use of a syntactic hierarchial ordering of the phrases was implemented. Any phrase that was a syntactic subphrase of another phrase was eliminated. The ordering was performed because the syntactic subphrase was a generalization of a broader phrase included in FIG. 7A.

WEST

Generate Collection | Print

L3: Entry 4 of 10 File: USPT Sep 11, 2001

US-PAT-NO: 6289501
DOCUMENT-IDENTIFIER: US 6289501 B1

TITLE: Method for generating simple document type definitions

DATE-ISSUED: September 11, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Mutschler, III; Eugene Otto	San Clemente	CA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Unisys Corp.	Blue Bell	PA			02

APPL-NO: 09/ 282345 [PALM]
DATE FILED: March 31, 1999

PARENT-CASE:
CROSS REFERENCE TO RELATED APPLICATIONS This patent document relates to the following patent applications, assigned to the same assignee hereof, which are incorporated herein by reference. U.S. Ser. No. 09/282,102, entitled A METHOD AND SYSTEM FOR GENERATING A COMPACT DOCUMENT TYPE DEFINITION FOR DATA INTERCHANGE AMONG SOFTWARE TOOLS; and, U.S. Ser. No. 09/282,230, entitled A METHOD AND SYSTEM FOR GENERATING A HIERARCHIAL DOCUMENT TYPE DEFINITION FOR DATA INTERCHANGE AMONG SOFTWARE TOOLS.

INT-CL: [07] G06 F 9/45

US-CL-ISSUED: 717/1; 707/513
US-CL-CURRENT: 717/114; 715/513

FIELD-OF-SEARCH: 717/1, 717/2, 717/3, 717/5, 717/8, 707/3, 707/4, 707/10, 707/100, 707/103, 707/513, 709/315, 709/316

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected | Search ALL

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	5761499	June 1998	Sonderegger	707/10
☐	5890158	March 1999	House et al.	707/10
☐	5970490	October 1999	Morgenstern	707/10
☐	6018627	January 2000	Iyengar et al.	395/701
☐	6167564	December 2000	Fontana et al.	717/1

FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY	US-CL
1004987	May 2000	EP	

OTHER PUBLICATIONS

Klely, "XML offers standard way of extending HTML", InformationWeek, Oct. 1997, pp. 8-12.*
Levin, "Component modeling tools encourage reuse", InformationWeek, Mar. 1997, pp 6-11.

ART-UNIT: 212

PRIMARY-EXAMINER: Chaki; Kakali

ABSTRACT:

A method is disclosed for use in a software development framework that has a repository and at least two software systems. The repository contains a meta-model and the software systems store instances of the meta-model. The method enables exchange of the instances of the meta-model among the software systems using a generalized data transfer language. The method comprises the steps of extracting a fixed component and a variable component of the metadata; parsing the variable component into constituent components; and, transforming each of the constituent components into corresponding components of a software language. The previous two steps are repeated for each instance of the variable component. Next, each instance of the variable component is transformed into corresponding components of the generalized software language. The fixed components are then transformed into corresponding components of the generalized software language. After this, the corresponding components are distributed among the software systems, where they are used as the format specification for the generalized data transfer language transport of the instances of the meta-model among the software systems.

15 Claims, 19 Drawing figures

L3: Entry 9 of 10 File: USPT Mar 2, 1999

DOCUMENT-IDENTIFIER: US 5878390 A
TITLE: Speech recognition apparatus equipped with means for removing erroneous candidate of speech recognition

Detailed Description Text (18):
Therefore, erroneous sentences are detected from the viewpoints of the matchability of semantic distances, as viewed generally, and the suitability of analyzed sentence constructions. More concretely, the judgment is carried out in the following way. First of all, mismatch of semantic distances in a partial sentence is judged by a semantic distance value that has been used in the above sentence construction determining method. If the total sum of semantic distances of a partial sentence is equal to or larger than a certain threshold value, the sentence is judged as a mis-recognition. The suitability of sentence construction is accounted as follows. It is assumed that a natural sentence composed of a certain or larger number of morphemes would have a united sentence construction, the syntactic structure of the sentence being complex to some extent. Now consider a ratio of the number m of morphemes contained in a partial sentence to the number R of rules of context-free grammar or rules of examples (or number of examples) that have been used for the determination of a sentence construction (i.e., a ratio of m/R). A partial sentence having no united sentence construction would not result in a hierarchical syntactic structure, so that the number R of syntax rules used is smaller relative to the number m of morphemes, with the value of m/R increased. Conversely, the more complex and the more hierarchical the sentence construction is, the smaller the m/R value is. Therefore, the following Equation (1) is defined as an erroneous sentence judging function F.sub.error (m): ##EQU1##

WEST

Generate Collection | Print

L3: Entry 9 of 10 File: USPT Mar 2, 1999

US-PAT-NO: 5878390
DOCUMENT-IDENTIFIER: US 5878390 A

TITLE: Speech recognition apparatus equipped with means for removing erroneous candidate of speech recognition

DATE-ISSUED: March 2, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Kawai; Jun	Osaka			JP
Wakita; Yumi	Nara			JP

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
ATR Interpreting Telecommunications Research Laboratories	Kyoto			JP	03

APPL-NO: 08/ 880403 [PALM]
DATE FILED: June 23, 1997

FOREIGN-APPL-PRIORITY-DATA:

COUNTRY	APPL-NO	APPL-DATE
JP	8-341084	December 20, 1996
JP	9-161243	June 18, 1997

INT-CL: [06] G10 L 9/00

US-CL-ISSUED: 704/231; 704/239, 704/256
US-CL-CURRENT: 704/231; 704/239, 704/256

FIELD-OF-SEARCH: 704/231, 704/239, 704/256, 704/255, 704/240, 704/243, 704/241

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected | Search ALL

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	5457768	October 1995	Tsuboi et al.	704/231
☐	5740318	April 1998	Naito et al.	704/248

OTHER PUBLICATIONS

Bahl et al., "A Maximum Likelihood Approach to Continuous Speech Recognition"; IEEE Transactions on Pattern Analysis and Machine Intelligence, Mar. 1983; pp. 179-190.
Eiichiro Sumita et al., "An Example-Based Disambiguation of English Prepositional Phrase Attachment"; 1995 Scripta Technica, Inc., Mar. 1994; pp. 557-565.
Oamu Furuse, "Transfer-Driven Machine Translation Utilizing Empirical Knowledge"; pp. 414-425.
Osamu Furuse, "Incremental Translatio Utilizing Constituent Boundary Patterns", ATR Interpreting Telecommunications Research Laboratories, pp. 412-417.

ART-UNIT: 271

PRIMARY-EXAMINER: Knepper; David D.

ASSISTANT-EXAMINER: Wieland; Susan

ABSTRACT:

A speech recognition apparatus which includes a speech recognition section for performing a speech recognition process on an uttered speech with reference to a predetermined statistical language model, based on a series of speech signal of the uttered speech sentence composed of a series of input words. The speech recognition section calculates a functional value of a predetermined erroneous sentence judging function with respect to speech recognition candidates, where the erroneous sentence judging representing a degree of unsuitability for the speech recognition candidates. When the calculated functional value exceeds a predetermined threshold value, the speech recognition section performs the speech recognition process by eliminating a speech recognition candidate corresponding to a calculated functional value.

12 Claims, 6 Drawing figures

WEST

End of Result Set

Generate Collection | Print

L3: Entry 10 of 10 File: USPT Apr 15, 1986

US-PAT-NO: 4583164
DOCUMENT-IDENTIFIER: US 4583164 A

TITLE: Syntactically self-structuring cellular computer

DATE-ISSUED: April 15, 1986

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Tolle; Donald M.	Chapel Hill	NC	27514	

APPL-NO: 06/ 294390 [PALM]
DATE FILED: August 19, 1981

INT-CL: [04] G06F 7/38, G06F 3/00, G06F 13/00

US-CL-ISSUED: 364/200; 364/300
US-CL-CURRENT: 712/11; 709/252, 717/143, 717/149

FIELD-OF-SEARCH: 364/200, 364/900, 364/300

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected | Search ALL

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	3962706	June 1976	Dennis et al.	364/200
☐	4075689	February 1978	Berkling	364/200
☐	4130885	December 1978	Dennis	364/200
☐	4149240	April 1979	Misunas et al.	364/200
☐	4251861	February 1981	Mago	364/200

OTHER PUBLICATIONS

International Journal of Computer & Information Sciences 8, 5 (Oct. 1979) and 8, 6 (Dec. 1979) "A Network of Microprocessors to Execute Reduction Languages" Mago.
Communications of the ACM 21, 8 (Aug. 1978) pp. 613-641 Backus "Can Programming be Liberated from the Von Neuman Style".
AFIPS Conference Proceedings, vol. 48, 1979 NCC Davis, "A Data Flow Evaluation System Based on the concept of Recursive Locality".

ART-UNIT: 237

PRIMARY-EXAMINER: Shaw; Gareth D.

ASSISTANT-EXAMINER: Lee; Jameson

ABSTRACT:

A design is disclosed for a cellular computer consisting of many processors, of two

kinds, connected in the form of a tree. The computer is intended for the highly parallel execution of programs written in an applicative programming language. The program is stored in the leaf cells of the tree. The computer uses the syntactic structure of the program to guide the embedding of a network of "syntactic nodes" in the tree of machine cells, and execution of the program is accomplished through operations performed by the embedded network of nodes. This computer can execute many user programs simultaneously, it can take advantage of all the parallelism expressed in each user program (storage space permitting), and it can perform in parallel many operations below the level expressed in the user programs.

8 Claims, 79 Drawing figures

End of Result Set

L3: Entry 10 of 10 File: USPT Apr 15, 1986

DOCUMENT-IDENTIFIER: US 4583164 A
TITLE: Syntactically self-structuring cellular computer

Brief Summary Text (130):
The embedded structure, corresponding as it does to the syntactic structure of the RA, and being distributed over a number of L cells and T cells, gives a natural allocation of computing resources, connected in a useful way. The problem of programming the nodes admits of a natural, hierarchical solution corresponding to the hierarchical syntactic structure of the RA.

CLAIMS:

1. A method for parallel execution of specified operations upon data having a hierarchical syntactic structure, said structure comprising expressions nested within expressions, said specified operations to be executed on specified ones of said expressions, comprising the steps of:

2. Forming a processing network by apportioning said computational resources to form, in said cells, nodes in correspondence with certain expressions of said data, and apportioning said communications resources to form channels connecting said nodes, said processing network comprising said nodes interconnected by said channels, said processing network having a hierarchical structure corresponding essentially to said hierarchical syntactic structure of said data, each said node comprising computational resources entirely within one of said cells and each said channel comprising communications resources within at least one of said cells, and connecting two said nodes, at least one of which comprises computational resources belonging to the same cell as do certain communications resources forming said channel, and

WEST

Generate Collection | Print

Search Results - Record(s) 1 through 10 of 10 returned.

☐ 1. Document ID: US 20020010714 A1

L3: Entry 1 of 10 File: PGPB Jan 24, 2002

PGPUB-DOCUMENT-NUMBER: 20020010714
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20020010714 A1

TITLE: Method and apparatus for processing free-format data

PUBLICATION-DATE: January 24, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY	RULE-47
Hetherington, Greg	Kareela		AU	

US-CL-CURRENT: 715/505; 715/508

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw Desc | Image

☐ 2. Document ID: US 6381743 B1

L3: Entry 2 of 10 File: USPT Apr 30, 2002

US-PAT-NO: 6381743
DOCUMENT-IDENTIFIER: US 6381743 B1

TITLE: Method and system for generating a hierarchial document type definition for data interchange among software tools

DATE-ISSUED: April 30, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Mutschler, III; Eugene Otto	San Clemente	CA		

US-CL-CURRENT: 717/104; 707/104.1, 717/101, 717/108, 717/116, 717/137

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw Desc | Image

☐ 3. Document ID: US 6308172 B1

L3: Entry 3 of 10 File: USPT Oct 23, 2001

US-PAT-NO: 6308172
DOCUMENT-IDENTIFIER: US 6308172 B1

TITLE: Method and apparatus for partitioning a database upon a timestamp, support values for phrases and generating a history of frequently occurring phrases

DATE-ISSUED: October 23, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Agrawal; Rakesh	San Jose	CA		
Srikant; Ramakrishnan	San Jose	CA		
Lent; Brian Scott	Union City	CA		

US-CL-CURRENT: 707/5; 704/4, 704/8, 704/9, 707/100, 707/102, 707/2, 707/203, 707/6, 715/511, 715/536

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments KWIC | Draw Desc | Image

---

☐ 4. Document ID: US 6289501 B1

L3: Entry 4 of 10 File: USPT Sep 11, 2001

US-PAT-NO: 6289501
DOCUMENT-IDENTIFIER: US 6289501 B1

TITLE: Method for generating simple document type definitions

DATE-ISSUED: September 11, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Mutschler, III; Eugene Otto	San Clemente	CA		

US-CL-CURRENT: 717/114; 715/513

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments KWIC | Draw Desc | Image

---

☐ 5. Document ID: US 6272495 B1

L3: Entry 5 of 10 File: USPT Aug 7, 2001

US-PAT-NO: 6272495
DOCUMENT-IDENTIFIER: US 6272495 B1

TITLE: Method and apparatus for processing free-format data

DATE-ISSUED: August 7, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Hetherington; Greg	Kareela New South Wales 2232			AU

US-CL-CURRENT: 707/101; 707/102, 707/4, 715/531

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments KWIC | Draw Desc | Image

---

☐ 6. Document ID: US 6253366 B1

L3: Entry 6 of 10 File: USPT Jun 26, 2001

US-PAT-NO: 6253366
DOCUMENT-IDENTIFIER: US 6253366 B1

TITLE: Method and system for generating a compact document type definition for data interchange among software tools

DATE-ISSUED: June 26, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Mutschler, III; Eugene Otto	San Clemente	CA		

US-CL-CURRENT: 717/104; 707/100, 717/107, 717/114

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments KWIC | Draw Desc | Image

---

☐ 7. Document ID: US 6006223 A

L3: Entry 7 of 10 File: USPT Dec 21, 1999

US-PAT-NO: 6006223
DOCUMENT-IDENTIFIER: US 6006223 A

TITLE: Mapping words, phrases using sequential-pattern to find user specific trends in a text database

DATE-ISSUED: December 21, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Agrawal; Rakesh	San Jose	CA		
Srikant; Ramakrishnan	San Jose	CA		
Lent; Brian Scott	Union City	WA		

US-CL-CURRENT: 707/5; 707/102, 707/203, 707/3, 707/6, 715/511, 715/532

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments KWIC | Draw Desc | Image

---

☐ 8. Document ID: US 5960384 A

L3: Entry 8 of 10 File: USPT Sep 28, 1999

US-PAT-NO: 5960384
DOCUMENT-IDENTIFIER: US 5960384 A

TITLE: Method and device for parsing natural language sentences and other sequential symbolic expressions

DATE-ISSUED: September 28, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Brash; Douglas E.	Killingworth	CT	06419	

US-CL-CURRENT: 704/9; 704/10, 704/235, 704/270

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments KWIC | Draw Desc | Image

---

☐ 9. Document ID: US 5878390 A

L3: Entry 9 of 10 File: USPT Mar 2, 1999

US-PAT-NO: 5878390
DOCUMENT-IDENTIFIER: US 5878390 A

TITLE: Speech recognition apparatus equipped with means for removing erroneous candidate of speech recognition

DATE-ISSUED: March 2, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Kawai; Jun	Osaka			JP
Wakita; Yumi	Nara			JP

US-CL-CURRENT: 704/231; 704/239, 704/256

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | KWIC | Draw Desc | Image

---

☐ 10. Document ID: US 4583164 A

L3: Entry 10 of 10 File: USPT Apr 15, 1986

US-PAT-NO: 4583164
DOCUMENT-IDENTIFIER: US 4583164 A

TITLE: Syntactically self-structuring cellular computer

DATE-ISSUED: April 15, 1986

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Tolle; Donald M.	Chapel Hill	NC	27514	

US-CL-CURRENT: 712/11; 709/252, 717/143, 717/149

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | KWIC | Draw Desc | Image

---

Generate Collection | Print

Terms	Documents
syntactic near hierarch$	10

Display Format: - Change Format

Previous Page Next Page